AMENDMENTS TO THE CLAIMS

1. (CURRENTLY AMENDED) A method in a server configured for executing web applications, the method comprising:

receiving a first ~~hypertext~~ markup language ~~(HTML)~~ request, via a ~~hypertext~~ transport ~~(HTTP)~~ connection, for a first ~~HTML~~ markup language page for a user;

generating the first ~~HTML~~ markup language page in response to the first ~~HTML~~ markup language request by executing a first web application instance according to a first application state;

storing a data record that specifies the first application state and a corresponding session identifier;

sending the first ~~HTML~~ markup language page and the session identifier to the user via the ~~HTTP~~ transport connection;

receiving via the ~~HTTP~~ transport connection a second ~~HTML~~ markup language request for a second ~~HTML~~ markup language page; and

generating the second ~~HTML~~ markup language page by selectively executing a second web application instance based on the first application state, based on reception of the corresponding session identifier in the second ~~HTML~~ markup language request; wherein:

the step of generating the first markup language page includes inserting a first voice web application parameter and terminating the first web application instance upon sending the first markup language page;

the step of generating the second markup language page includes (1) initiating the second web application instance, following termination of the first web application instance, in response to the selection thereof based on identifying the first application state specified in the data record, and (2) inserting a second voice web application parameter, the first and second markup language pages representing respective application states of a web-enabled voice messaging session with the user.

2. (ORIGINAL) The method of claim 1, wherein the storing step includes storing user attribute information that specifies attributes about the user in the data record.

3. (CURRENTLY AMENDED) The method of claim 2, wherein the step of generating the second ~~HTML~~ markup language page includes executing the second web application instance based on the user attribute information stored in the corresponding data record.

4. (ORIGINAL) The method of claim 3, wherein the storing step includes storing in the user attribute information subscriber profile information that specifies profile and preference settings for the corresponding user.

5. (ORIGINAL) The method of claim 2, further comprising deleting the data record after a prescribed interval.

6. (ORIGINAL) The method of claim 2, wherein the storing step further includes storing the data record as an extensible markup language (XML) document.

7. (ORIGINAL) The method of claim 6, wherein the storing step further includes storing the XML document in a nonvolatile memory locally within the server.

8. (ORIGINAL) The method of claim 7, further comprising forwarding the XML document from the nonvolatile memory to a second server requesting the XML document.

9. (CURRENTLY AMENDED) The method of claim 6, wherein the storing step further includes storing the XML document in a nonvolatile memory of a second server in communication with the server, the steps of generating the first and second ~~HTML~~ markup language pages each comprising registering with the second server for corresponding access to the XML document.

10. (ORIGINAL) The method of claim 1, wherein the storing step includes:

registering the first web application instance with a registry and in response creating a registry entry; and

storing state attributes, generated during execution of the first web application instance and describing the first application state, into the registry entry.

11. (CURRENTLY AMENDED) The method of claim 1, wherein the step of sending the first ~~HTML~~ markup language page and the session identifier includes adding a tag within the first ~~HTML~~ markup language page that includes a uniform resource locator (URL) that specifies the session identifier.

12. (CURRENTLY AMENDED) The method of claim 11, wherein the step of generating the second ~~HTML~~ markup language page includes detecting the session identifier within the URL supplied by the second ~~HTML~~ markup language request.

13. (CURRENTLY AMENDED) The method of claim 1, wherein the step of sending the first ~~HTML~~ markup language page and the session identifier includes sending a cookie that includes the session identifier.

14. (CURRENTLY AMENDED) The method of claim 13, wherein the step of generating the second ~~HTML~~ markup language page includes detecting the session identifier within a cookie supplied with the second ~~HTML~~ markup language request.

15. (CURRENTLY AMENDED) A computer readable medium having stored thereon sequences of instructions for executing web applications by a server, the sequences of instructions including instructions for performing the steps of:

receiving a first ~~hypertext~~ markup language ~~(HTML)~~ request, via a ~~hypertext~~ transport

~~(HTTP)~~ connection, for a first ~~HTML~~ <u>markup language</u> page for a user;

generating the first ~~HTML~~ <u>markup language</u> page in response to the first ~~HTML~~

<u>markup language</u> request by executing a first web application instance according to a first

application state;

storing a data record that specifies the first application state and a corresponding session

identifier;

sending the first ~~HTML~~ <u>markup language</u> page and the session identifier to the user via

the ~~HTTP~~ <u>transport</u> connection;

receiving via the ~~HTTP~~ <u>transport</u> connection a second ~~HTML~~ <u>markup language</u> request

for a second ~~HTML~~ <u>markup language</u> page; and

generating the second ~~HTML~~ <u>markup language</u> page by selectively executing a second

web application instance based on the first application state, based on reception of the

corresponding session identifier in the second ~~HTML~~ <u>markup language</u> request<u>; wherein:</u>

<u>the step of generating the first markup language page includes inserting a first voice web</u>

<u>application parameter and terminating the first web application instance upon sending the first</u>

<u>markup language page;</u>

<u>the step of generating the second markup language page includes (1) initiating the second</u>

<u>web application instance, following termination of the first web application instance, in response</u>

<u>to the selection thereof based on identifying the first application state specified in the data record,</u>

<u>and (2) inserting a second voice web application parameter, the first and second markup language</u>

pages representing respective application states of a web-enabled voice messaging session with the user.

16. (ORIGINAL) The medium of claim 15, wherein the storing step includes storing user attribute information that specifies attributes about the user in the data record.

17. (CURRENTLY AMENDED) The medium of claim 16, wherein the step of generating the second ~~HTML~~ markup language page includes executing the second web application instance based on the user attribute information stored in the corresponding data record.

18. (ORIGINAL) The medium of claim 17, wherein the storing step includes storing in the user attribute information subscriber profile information that specifies profile and preference settings for the corresponding user.

19. (ORIGINAL) The medium of claim 16, further comprising instructions for performing the step of deleting the data record after a prescribed interval.

20. (ORIGINAL) The medium of claim 16, wherein the storing step further includes storing the data record as an extensible markup language (XML) document.

21. (ORIGINAL) The medium of claim 20, wherein the storing step further includes storing the XML document in a nonvolatile memory locally within the server.

22. (ORIGINAL) The medium of claim 21, further comprising instructions for performing the step of forwarding the XML document from the nonvolatile memory to a second server requesting the XML document.

23. (CURRENTLY AMENDED) The medium of claim 20, wherein the storing step further includes storing the XML document in a nonvolatile memory of a second server in communication with the server, the steps of generating the first and second ~~HTML~~ markup language pages each comprising registering with the second server for corresponding access to the XML document.

24. (ORIGINAL) The medium of claim 15, wherein the storing step includes:

registering the first web application instance with a registry and in response creating a registry entry; and

storing state attributes, generated during execution of the first web application instance and describing the first application state, into the registry entry.

25. (CURRENTLY AMENDED) The medium of claim 15, wherein the step of sending the first ~~HTML~~ markup language page and the session identifier includes adding a tag within

the first ~~HTML~~ markup language page that includes a uniform resource locator (URL) that

specifies the session identifier.

26. (CURRENTLY AMENDED) The medium of claim 25, wherein the step of

generating the second ~~HTML~~ markup language page includes detecting the session identifier

within the URL supplied by the second ~~HTML~~ markup language request.

27. (CURRENTLY AMENDED) The medium of claim 15, wherein the step of sending

the first ~~HTML~~ markup language page and the session identifier includes sending a cookie that

includes the session identifier.

28. (CURRENTLY AMENDED) The medium of claim 27, wherein the step of

generating the second ~~HTML~~ markup language page includes detecting the session identifier

within a cookie supplied with the second ~~HTML~~ markup language request.

29. (CURRENTLY AMENDED) A processor-based system configured for executing

web applications, the device comprising:

a ~~hypertext~~ transport protocol ~~(HTTP)~~ interface configured receiving first and second

markup language requests for first and second ~~HTML~~ markup language pages for a user,

respectively, and sending the first and second ~~HTML~~ markup language pages and a session

identifier to the user via ~~an HTTP~~ a transport connection; and

an application server configured for executing first and second web application instances for generation of the first and second ~~HTML~~ markup language pages in response to the first and second ~~HTML~~ markup language requests, respectively, the application server storing a data record that specifies a session state with the user upon completion of the first web application instance, the application server accessing the data record in response to detecting the session identifier in the second ~~HTML~~ markup language request, and executing the second web application instance based on the accessed data record wherein:

the application server is configured for terminating the first web application instance upon the sending of the first markup language page,

the application server configured for initiating the second web application instance, following termination of the first web application instance, in response to the selection thereof based on identifying the session state specified in the data record,

the application server configured for inserting first and second voice web application parameters into the first and second markup language pages during execution of the first and second web application instances, respectively, the first and second markup language pages representing respective application states of a web-enabled voice messaging session with the user.

30. (CURRENTLY AMENDED) The system of claim 29, wherein the ~~HTTP~~ transport interface includes a web server connected to an Internet Protocol (IP) network.

31. (ORIGINAL) The system of claim 29, further including a local memory for storing the data record for a prescribed time interval.

32. (ORIGINAL) The system of claim 29, further comprising a shared registry for storing the data record, the shared registry configured for supplying the data record to authorized servers.

33. (ORIGINAL) The system of claim 29, wherein the application server is configured for storing within the data record user attribute information that specifies attributes about the user, the application server executing the second web application instance based on the user attribute information in the corresponding accessed data record.

34. (ORIGINAL) The system of claim 33, wherein the application server stores the data record as an extensible markup language (XML) document.

35. (CURRENTLY AMENDED) The system of claim 29, wherein the application server adds to the first ~~HTML~~ markup language page a tag that includes a uniform resource locator (URL) that specifies the session identifier, the application server accessing the data record in response to detecting the URL specifying the session identifier in the second ~~HTML~~ markup language request.

36. (CURRENTLY AMENDED) The device of claim 29, wherein the application server sends with the first ~~HTML~~ markup language page a cookie that includes the session identifier, the application server accessing the data record in response to detecting the cookie having the session identifier with the second ~~HTML~~ markup language request.

37. (NEW) The method of claim 1, wherein the transport connection is a hypertext transport (HTTP) connection and the first and second voice web application parameters each include extensible markup language (XML) based parameters.

38. (NEW) The medium of claim 15, wherein the transport connection is a hypertext transport (HTTP) connection and the first and second voice web application parameters each include extensible markup language (XML) based parameters.

39. (NEW) The system of claim 29, wherein the transport interface is a hypertext transport (HTTP) interface and the transport connection is an HTTP connection, the first and second voice web application parameters each including extensible markup language (XML) based parameters.

40. (NEW) A method in a communications system, the method comprising:

receiving by a proxy browser a request from a user device via a device connection

according to a corresponding device protocol, and in response generating in the proxy browser a

first markup language request;

outputting the first markup language request by the proxy browser to a server via a

transport connection;

receiving by the server the first markup language request via the transport connection, the

first markup language request requesting a first markup language page for a user of the user

device;

generating by the server the first markup language page in response to the first markup

language request by executing a first web application instance according to a first application

state;

storing by the server a data record that specifies the first application state and a

corresponding session identifier;

sending by the server the first markup language page and the session identifier to the

proxy browser via the transport connection;

receiving by the proxy browser the first markup page and the session identifier via the

transport connection, and in response providing selected media content specified in the first

markup page to the user device via the device connection, the selected media content selected

based on the device protocol;

receiving by the proxy browser a second request from a user device via the device

connection according to the corresponding device protocol, and in response generating in the

proxy browser a second markup language request;

outputting the second markup language request, including the session identifier, by the

proxy browser to the server via the transport connection;

receiving by the server via the transport connection the second markup language request

for the second markup language page and including the session identifier; and

generating by the server the second markup language page by selectively executing a

second web application instance based on the first application state, based on reception of the

corresponding session identifier in the second markup language request.

41. (NEW) The method of claim 40, wherein the storing step includes storing user

attribute information that specifies attributes about the user in the data record.

42. (NEW) The method of claim 41, wherein the step of generating the second markup

language page includes executing the second web application instance based on the user attribute

information stored in the corresponding data record.

43. (NEW) The method of claim 42, wherein the storing step includes storing in the user

attribute information subscriber profile information that specifies profile and preference settings

for the corresponding user.

44. (NEW) The method of claim 41, further comprising deleting the data record by the server after a prescribed interval.

45. (NEW) The method of claim 41, wherein the storing step further includes storing by the server the data record as an extensible markup language (XML) document.

46. (NEW) The method of claim 45, wherein the storing step further includes storing the XML document in a nonvolatile memory locally within the server.

47. (NEW) The method of claim 46, further comprising forwarding by the server the XML document from the nonvolatile memory to a second server requesting the XML document.

48. (NEW) The method of claim 45, wherein the storing step further includes storing the XML document in a nonvolatile memory of a second server in communication with the server, the steps of generating the first and second markup language pages each comprising registering with the second server for corresponding access to the XML document.

49. (NEW) The method of claim 40, wherein the storing step includes:

registering the first web application instance with a registry and in response creating a registry entry; and

storing state attributes, generated during execution of the first web application instance and describing the first application state, into the registry entry.

50. (NEW) The method of claim 40, wherein the step of sending the first markup language page and the session identifier includes adding a tag within the first markup language page that includes a uniform resource locator (URL) that specifies the session identifier.

51. (NEW) The method of claim 40, wherein the step of sending the first markup language page and the session identifier includes sending a cookie that includes the session identifier.

52. (NEW) The method of claim 40, wherein:

the step of generating the first markup language page includes inserting a first voice web application parameter and terminating the first web application instance upon sending the first markup language page;

the step of generating the second markup language page includes (1) initiating the second web application instance, following termination of the first web application instance, in response to the selection thereof based on identifying the first application state specified in the data record, and (2) inserting a second voice web application parameter, the first and second markup language pages representing respective application states of a web-enabled voice messaging session with the user.

53. (NEW) The method of claim 40, wherein the transport connection is a hypertext transport protocol (HTTP) connection, and the first markup language page is a Hypertext Markup Language (HTML) page.

54. (NEW) A system comprising:

a proxy browser configured for receiving first and second requests from a user device via a device connection according to a corresponding device protocol and in response generating in the proxy browser respective first and second markup language requests; and

a server comrising:

(1) a transport protocol interface configured receiving the first and second markup language requests for respective first and second markup language pages for a user of the user device, and sending the first and second markup language pages and a session identifier to the proxy browser via a transport connection, and

(2) an application server configured for executing first and second web application instances for generation of the first and second markup language pages in response to the first and second markup language requests, respectively, the application server storing a data record that specifies a session state with the user upon completion of the first web application instance, the application server accessing the data record in response to detecting the session identifier in the second markup language request, and executing the second web application instance based on the accessed data record.

55. (NEW) The system of claim54, wherein the transport interface includes a web server connected to an Internet Protocol (IP) network.

56. (NEW) The system of claim 54, wherein the server further includes a local memory for storing the data record for a prescribed time interval.

57. (NEW) The system of claim 54, wherein the server further includes a shared registry for storing the data record, the shared registry configured for supplying the data record to authorized servers.

58. (NEW) The system of claim 54, wherein the application server is configured for storing within the data record user attribute information that specifies attributes about the user, the application server executing the second web application instance based on the user attribute information in the corresponding accessed data record.

59. (NEW) The system of claim 58, wherein the application server stores the data record as an extensible markup language (XML) document.

60. (NEW) The system of claim 54, wherein the application server adds to the first markup language page a tag that includes a uniform resource locator (URL) that specifies the

session identifier, the application server accessing the data record in response to detecting the URL specifying the session identifier in the second markup language request.

61. (NEW) The system of claim 54, wherein the application server sends with the first markup language page a cookie that includes the session identifier, the application server accessing the data record in response to detecting the cookie having the session identifier with the second markup language request.

62. (NEW) The system of claim 54, wherein:

the application server is configured for terminating the first web application instance upon the sending of the first markup language page,

the application server configured for initiating the second web application instance, following termination of the first web application instance, in response to the selection thereof based on identifying the session state specified in the data record,

the application server configured for inserting first and second voice web application parameters into the first and second markup language pages during execution of the first and second web application instances, respectively, the first and second markup language pages representing respective application states of a web-enabled voice messaging session with the user.